UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/900,494	07/06/2001	Michael Freed	NEXSI-01112US0	4136

28863 7590 05/10/2007

SHUMAKER & SIEFFERT, P. A.
1625 RADIO DRIVE
SUITE 300
WOODBURY, MN 55125

EXAMINER	
MOORTHY, ARAVIND K	

ART UNIT	PAPER NUMBER
2131	

MAIL DATE	DELIVERY MODE
05/10/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/900,494
Filing Date: July 06, 2001
Appellant(s): FREED ET AL.

Jennifer M.K. Rogers
Reg. No. 58,695
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 27 December 2006 appealing from the Office action mailed 29 June 2006.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6799202	Gelman et al	07-2002
6654344	Toporek et al	11-2003
6732269	Baskey et al	05-2004
6732175	Abjanic	05-2004
6799202	Hankinson et al	09-2004

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 112

The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 1-28 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

Regarding independent claim 1, the claim has been amended to include the limitation "wherein the decryption engine and the load balancing engine bypass an application layer of a network stack by decrypting the data from the secure communication sessions of the clients and outputting the decrypted data to the associated server devices without processing the data with the application layer of the network stack". There is no support for this limitation in the

specification. In the recently filed arguments, the applicant argues that figure 3 of the current application discloses this limitation. However, the examiner finds no support in figure 3 or the detailed explanation of figure 3 for this limitation. The examiner requests the applicant to specifically point out where in the specification this limitation is taught.

Regarding independent claims 12 and 25, both the claims have been amended to include the limitation "without processing the data packets with an application layer of a network stack". There is no support for this limitation in the specification. In the recently filed arguments, the applicant argues that figure 3 of the current application discloses this limitation. However, the examiner finds no support in figure 3 or the detailed explanation of figure 3 for this limitation. The examiner requests the applicant to specifically point out where in the specification this limitation is taught.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-7 and 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hankinson et al U.S. Patent No. 6,799,202 B1 in view of Toporek et al U.S. Patent No. 6,654,344 B1.

As to claim 1, Hankinson et al discloses a load balancing acceleration device, comprising:

a processor, memory and communications interface [column 5, lines 26-48];

a TCP communications manager capable of interacting with a plurality of client devices and server devices simultaneously via the communications interface [column 20, lines 11-29];

a secure communications manager to negotiate a secure communication session with one of the client devices [column 8 line 49 to column 9 line 14];

an encryption and decryption engine instructing the processor to decrypt data received via the secure communications session and direct the decrypted data it to one of said server devices via a second communication session [column 8 line 49 to column 9 line 14]; and

a load balancing engine associating each of said client devices with a respective one of said servers devices based on calculated processing loads of each said server devices [column 17 line 41 to column 18 line 24].

Hankinson et al does not teach that the decryption engine and the load balancing engine bypass an application layer of a network stack by decrypting the data from the secure communication sessions of the clients and outputting the decrypted data to the associated server devices without processing the data with the application layer of the network stack.

Toporek et al teaches bypassing an application layer of a network stack [column 11, lines 22-33].

Therefore, it would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Hankinson et al so that the decryption engine and the load balancing engine would have bypassed an application layer of a network stack by decrypting the data from the secure communication sessions of the clients and outputting the decrypted data to the associated server devices without processing the data with the application layer of the network stack.

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Hankinson et al by the teaching of Toporek et al because it allows the network layer to communicate directly with the physical layer [column 11, lines 22-33].

As to claim 2, Hankinson et al teaches that the TCP communications manager provides an IP address of an enterprise to said secure communications manager, and each of said plurality of server devices is associated with the enterprise [column 7, lines 14-40].

As to claim 3, Hankinson et al teaches that the secure communications manager negotiates a secure communication session with each of said plurality of client devices over an open network [column 8 line 49 to column 9 line 14].

As to claim 4, Hankinson et al teaches that the TCP communications manager negotiates a separate, open communications session with one of the plurality of server devices associated with the enterprise for each secure communications session negotiated with the client devices based on the associations of said client devices to said server devices said load balancing engines [column 17 line 41 to column 18 line 24].

As to claim 5, Hankinson et al teaches that the encryption and decryption engine decrypts the data on a packet level by decrypting packet data received on the communications interface via the secure communications session to extract a secure record [column 13, lines 17-30]. Hankinson et al teaches decrypting application data from the secure record in the packet data [column 13, lines 17-30]. Hankinson et al teaches outputting the decrypted application data from the secure record to the one of said server devices via the second communication session without processing the application data with the application layer of the network stack [column 13, lines 17-30].

As to claim 6, Hankinson et al teaches that the load-balancing engine selects the second communication session [column 17 line 41 to column 18 line 24].

As to claim 7, Hankinson et al teaches that the TCP communications manager responds to TCP communications negotiations directly for an enterprise [column 12 line 65 to column 13 line 30].

As to claim 22, Hankinson et al teaches that the device comprises a network router [column 12 line 65 to column 13 line 30].

Claims 12-15, 17-21, 23 and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Abjanic U.S. Patent No. 6,732,175 B1 in view of Toporek et al U.S. Patent No. 6,654,344 B1.

As to claim 12, Abjanic discloses a method for performing acceleration of data communications between a plurality of customer devices attempting to communicate with an enterprise having a plurality of servers, comprising:

providing an intermediate acceleration device enabled for secure communication with the customer devices, wherein the acceleration device has an IP address associated with the enterprise [column 10, lines 1-32];

receiving with the acceleration device communications directed to the enterprise in a secure protocol from one of the customer devices [column 10, lines 1-32];

decrypting data packets of the secure protocol with the acceleration device to provide decrypted packet data [column 10, lines 1-32];

selecting with the acceleration device at least one of the plurality of servers in the enterprise based on a load calculation including processing sessions of other servers in the enterprise and associating the selected server with a communications session from the one of the clients [column 10, lines 1-32]; and

forwarding the decrypted packet data from the acceleration device to the selected server of the enterprise [column 10, lines 1-32].

Abjanic does not teach without processing the data packets with an application layer of a network stack.

Toporek et al teaches bypassing an application layer of a network stack [column 11, lines 22-33].

Therefore, it would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Abjanic so that the data packets within an application layer of a network stack would not have been processed.

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Abjanic by the teaching of Toporek et al because it allows the network layer to communicate directly with the physical layer [column 11, lines 22-33].

As to claim 13, Abjanic teaches the steps of receiving application data from the selected server of the enterprise, encrypting the application data received from the selected server, and forwarding encrypted application data to the customer device [column 10, lines 1-32].

As to claim 14, Abjanic teaches that the step of receiving communications directed to the enterprise includes receiving with the device communications having a destination IP address of the enterprise [column 10, lines 1-32].

As to claim 15, Abjanic teaches the step of negotiating the secure protocol session with the customer device by responding as the enterprise to the customer devices [column 10, lines 33-67].

As to claim 17, Abjanic teaches that the step of forwarding comprises:

establishing an open communication session from the acceleration device to the selected server [column 5, lines 14-40], and

mapping the decrypted packet data to the open communication session established with the selected server [column 5, lines 14-40].

As to claim 18, Abjanic teaches that the open communication session is established via a secure network [column 5, lines 14-40].

As to claim 19, Abjanic teaches that the step of receiving comprises:

receiving encrypted data having a length greater than a TCP segment carrying said data [column 10, lines 1-32]; and

wherein said step of decrypting comprises:

buffering the encrypted data in a memory buffer in the acceleration device, the buffer having a length equivalent to the block cipher size necessary to perform the cipher [column 10, lines 1-32]; and

decrypting the buffered segment of the received encrypted data to provide decrypted application data [column 10, lines 1-32].

As to claim 20, Abjanic teaches the step of authenticating the data on receipt of a final TCP segment on a packet level without processing the application data with an application layer of a TCP/IP stack [column 6, lines 1-27].

As to claim 21, Abjanic teaches the step of generating an alert if said step of authenticating results in a failure [column 6 line 63 to column 7 line 12].

As to claim 23, Abjanic teaches decrypting data packets comprises decrypting the data packets at a packet level of the network stack [column 6, lines 1-27].

As to claim 24, Abjanic teaches that decrypting data packets comprises:

decrypting the data packets to extract a secure record [column 10, lines 1-32],

decrypting application data from the secure record [column 10, lines 1-32], and

authenticating the application data without processing the application data with an application layer of a TCP/IP stack [column 10, lines 1-32].

Claims 25-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Baskey et al U.S. Patent No. 6,732,269 B1 in view of Toporek et al U.S. Patent No. 6,654,344 B1.

As to claim 25, Baskey et al discloses a system comprising:

a client device [column 5, lines 17-57];

a plurality of server devices [column 5, lines 17-57]; and

an intermediate device coupled between the client devices and the server devices [column 5, lines 17-57],

wherein the intermediate device intercepts a request from the client device for a secure communication session [column 5 line 58 to column 6 line 16], and

wherein, in response to the request, the intermediate device establishes a secure communication session with the client device, selects one of the server devices based on resource loading experienced by the server devices, and

establishes a non-secure communication session with the selected server device [column 5 line 58 to column 6 line 16].

Baskey et al does not teach without processing the data packets with an application layer of a network stack.

Toporek et al teaches bypassing an application layer of a network stack [column 11, lines 22-33].

Therefore, it would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Baskey et al so that the data packets within an application layer of a network stack would not have been processed.

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Baskey et al by the teaching of Toporek et al because it allows the network layer to communicate directly with the physical layer [column 11, lines 22-33].

As to claim 26, Baskey et al teaches that the intermediate device receives encrypted data from the client device via the secure communication session, decrypts the data and forwards the decrypted data to the selected server device via the non-secure communication session [column 8 line 51 to column 9 line 19].

As to claim 27, Baskey et al teaches that the intermediate device receives unencrypted data from the selected server device via the non-secure communication session, encrypts the data and forwards the encrypted data to the client device via the secure communication session [column 8 line 51 to column 9 line 19].

As to claim 28, Baskey et al teaches that the intermediate device comprises a network router [column 5, lines 17-57].

Claims 8-11 is rejected under 35 U.S.C. 103(a) as being unpatentable over Hankinson et al U.S. Patent No. 6,799,202 B1 and Toporek et al U.S. Patent No. 6,654,344 B1 as applied to claim 1 above, and further in view of Gelman et al U.S. Patent No. 6,415,329 B1.

As to claims 8 and 11, the Hankinson-Toporek combination does not teach that the secure communications manager changes a destination IP address for each packet to a server IP address for each session.

Gelman et al teaches a secure communications manager that changes a destination IP address for each packet to a server IP address for each session [column 10, lines 9-21].

Therefore, it would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified the Hankinson-Toporek combination so that the proxy server would have changed the destination IP address for each packet to one of the server IP addresses for each session.

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified the Hankinson-Toporek combination by the teaching of Gelman et al because the detrimental effects of latency and errors on TCP are avoided and link utilization is greatly increased. TCP/IP headers are replaced with a much shorter WLP header, leaving more bandwidth for data. In addition, TCP/IP data may be compressed so that fewer bytes need to be sent over the wireless segment, thus improving data transfer times. Encryption may also be used to protect data from eavesdropping. Finally, the system may be implemented

without making any changes to the TCP/IP code on the gateway. No changes of any kind are required to the end users [column 5, lines 54-67].

As to claim 9, the Hankinson-Toporek combination teaches that the TCP communications manager maintains TCP communication sessions with the server devices, and wherein the secure communications manager engine negotiates a secure communication session for each TCP communications session [Hankinson et al column 8 line 49 to column 9 line 14].

As to claim 10, the Hankinson-Toporek combination teaches that the secure communications manager responds to all secure communications with each client device [Hankinson et al column 8 line 49 to column 9 line 14].

Claim 16 is rejected under 35 U.S.C. 103(a) as being unpatentable over Abjanic U.S. Patent No. 6,732,175 B1 and Toporek et al U.S. Patent No. 6,654,344 B1 as applied to claim 12 above, and further in view of Gelman et al U.S. Patent No. 6,415,329 B1.

As to claim 16, the Abjanic-Toporek combination does not teach that the step of forwarding comprises modifying the destination IP address of data packets from the enterprise IP to an IP for the selected server.

Gelman et al teaches a secure communications manager that changes a destination IP address for each packet to a selected server IP address for each session [column 10, lines 9-21].

Therefore, it would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified the Abjanic-Toporek combination so that the proxy server would have changed the destination IP address for each packet to one of the server IP addresses for each session.

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified the Abjanic-Toporek combination by the teaching of Gelman et al because the detrimental effects of latency and errors on TCP are avoided and link utilization is greatly increased. TCP/IP headers are replaced with a much shorter WLP header, leaving more bandwidth for data. In addition, TCP/IP data may be compressed so that fewer bytes need to be sent over the wireless segment, thus improving data transfer times. Encryption may also be used to protect data from eavesdropping. Finally, the system may be implemented without making any changes to the TCP/IP code on the gateway. No changes of any kind are required to the end users [column 5, lines 54-67].

(10) Response to Argument

A. *On page 10, the Appellants argue that the specification describes the claimed subject matter in such a way to enable one skilled in the art to make and use the invention. Specifically, with respect to claim 1, the specification described the claimed subject matter in such a way to enable one skilled in the art to make and use a load balancing acceleration device that bypasses and application layer of a network stack by decrypting the data from the secure communications sessions of the clients and outputting the decrypted data to the associated server devices without processing the data with the application layer of the network stack.*

The examiner respectfully disagrees. On pages 10-12 of the appeal brief, the Appellants first refer to figure 2B and pages 5 and 6 of the present application that describe prior art SSL acceleration devices. The prior art SSL acceleration device of figure 2B receives encrypted SSL packets from the web client and processes the data up the network stack through the session layer all the way to the application layer (HTTP), i.e., layer seven of the OSI network stack. The prior

art SSL accelerator then processes data back down the network stack to forward decrypted packets to the web server.

After describing the prior art, the Appellants argue (on page 12 of the appeal brief) that the present application then describes in detail an acceleration device that decrypts the data from the secure communication sessions of the clients and outputs the decrypted data to the associated server devices without processing the data with the application layer of the network stack. The Appellants argue (on page 13 of the appeal brief) that page 10 of the present application specifically states that "rather than transmitting packets up and down the TCP/IP stack as show in figure 2B, [the SSL accelerator of Figure 3] will perform the SSL encryption and decryption at the packet level before forwarding the packet on to its destination."

The examiner acknowledges that page 10 of the present application specifically states that "rather than transmitting packets up and down the TCP/IP stack as show in figure 2B, [the SSL accelerator of Figure 3] will perform the SSL encryption and decryption at the packet level before forwarding the packet on to its destination." The examiner points out to the board specifically in that statement that SSL encryption and decryption is being performed. The examiner would also like to point out to the board that one of ordinary skill in the art knows that SSL is performed in the application layer. Therefore, if SSL encryption and decryption is being performed, it has to be accomplished in the application layer.

The Appellants continue to argue (on page 14 of the appeal brief) that there are three modes that are described in great detail of pages 13-30 of the present application, including description of detailed flow charts showing how these various modes implement SSL within the accelerator without requiring that the application data be reassembled at the application layer.

The examiner asserts that all three modes as described by the Appellants in the present application incorporate SSL. The examiner emphasizes that SSL occurs in the application layer. The examiner asserts that the Appellants have not provided any details in the specification how SSL is being performed outside of the application layer.

B. *On page 17, the Appellants argue that Hankinson in view of Toporek fails to establish a prima facie case of obviousness.*

The examiner respectfully disagrees. In response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

C. *The Appellants argues that there is no teaching in Hankinson in view of Toporek as to how a device could decrypt secure data using a process that bypasses the application layer. The Appellants argues that the combination of references provides no teaching as to how the application layer could be avoided for functions like SSL that traditionally require the application layer.*

The examiner points out to the board that none of the independent claims recite the use of SSL but merely using a secure communications. However, Hankinson teaches the use of SSL with encryption/decryption. Hankinson teaches the Federated OS is implemented with portable source code, which permits supporting heterogeneous CPU hardware, thereby allowing the freedom to choose from different processors and different vendors. This portability facilitates optimizing members for specific functionality. For example, an implementation of a member can

be optimized to use DSP (Digital Signal Processor) based encryption/decryption engines to support SSL (Secure Sockets Layer) or other secure protocols. Preferably, dependencies on a processor's native byte order, word size, etc., are encapsulated in a minimal code module for each type of processor [column 8, lines 49-60].

In addition, Toporek teaches bypassing the transport and application layers altogether. Toporek teaches information can flow through a gateway, such as gateway 203, 205 at the network layer, bypassing the transport and application layers altogether. For example, in gateway 203, information can enter from client 201 via telecommunications link 221. Physical layer 223 passes the information to driver 225. In turn, driver 225 passes the information to network layer 227. Network layer 227 can route the information to its destination using IP. The information then flows from network layer 227 to driver 235. Driver 235 interacts with physical layer 237 to pass the information along to satellite 209 via telecommunications link 239 [Toporek column 11, lines 22-33]. Therefore, the combination of Hankinson of Toporek would have a secure communications (i.e. SSL) with encryption and decryption. During transmission, the data would bypass the application and transport layers.

D. *The Appellants argue that the examiner's conclusion of obviousness is based entirely on the unsupported assumption that the Hankinson in view of Toporek could somehow achieve an acceleration device capable of decrypting secure client communications without processing the secure communication at the application layer.*

The Hankinson reference teaches that the architecture of the Federated OS inherently implements load balancing. Hankinson teaches that the architecture of the Federated OS inherently implements load balancing, because the load is inherently distributed across the

members. For example, in a TCP/IP embodiment, the TCP/IP state machine is distributed across a plurality of members. Preferably, the Federated OS also includes intelligent load balancing for dynamically assigning resources to match changing user demand. Dynamic load balancing can be accomplished, for example, by having a dispatcher assign tasks to the responder that the dispatcher determines is most lightly loaded. Alternatively, or in addition to dynamic load balancing controlled by dispatchers, dynamic load balancing can be accomplished by having responders determine when they are underutilized. When a dispatcher and/or responder(s) determine that one or more responders are lightly loaded, the functionality of two or more responders can be consolidated on a smaller number of responders (in a manner similar to that used for recovering from a failed member). The class of the resulting unused member(s) can be changed by restarting the unused member(s) as members of member classes that are experiencing a greater load, for example, receivers. Static load balancing can be simply implemented by dividing the data among the responders. Another approach for load balancing is a geographic algorithm, in which the responder that is geographically closest to the requesting client is selected. Yet another approach for load balancing is a network topology algorithm in which the responder that is closest to the client in terms of network topology is selected. The responders loads can also be distributed based on the capabilities of each responder. For example, responses requiring encryption capabilities are assigned to responders that have encryption capabilities. Load balancing can be automated, or can require human intervention [column 17 line 41 to column 18 line 6].

One of ordinary skill in the art would realize that load balancing accelerates a network. As taught by Hankinson, computing and distribution at the application level results in latencies

and other difficulties. Therefore, by bypassing the application layer altogether the result would be increased speed, security, reliability, scalability, capacity, and cost effectiveness, that also has reduced space, power, and cooling requirements, as well as reduced maintenance and operating costs [Hankinson column 2, lines 3-16].

E. *The Appellants argue that the Hankinson federated operating system is not load balancing client devices with respect to server devices, as required by claim 1 in the present application.*

As discussed above, Hankinson teaches load balancing. Hankinson teaches a server implementing a Federated OS preferably includes at least one receiver member, at least one dispatcher member, and at least one responder member (and preferably also includes one or more additional members in other member classes). In an alternative embodiment, a dispatcher member is not included, and the functionality of a dispatcher member is implemented on a receiver member and/or a responder member (or could be implemented on another member, from another member class). Embodiments that are coupled to an external network include at least one member that is coupled to the external network (and preferably include at least one receiver and at least one responder that are coupled to the external network). In embodiments that are not coupled to an external network, a receiver member and/or a responder member need not be included. The number of members in a server, and the classes of the members in a server, are determined by the services that are to be provided by the server, the load on the system, human intervention, and other factors [column 6, lines 20-38]. Although not required, preferably, every member hardware unit has the capability to be dynamically reconfigured during operation ("on the fly") to perform the function of any non-abstract member. In other words, the CPU(s) of a member preferably can be dynamically assigned any of the member functions. For example, if a

dispatcher becomes inoperable, a responder or receiver, for example, could be dynamically reconfigured during operation of a server to function as a dispatcher. This capability also permits dynamic load balancing. Dynamically reconfiguring member hardware units permits fault recovery without loss of service. Thus, a server implementing the Federated OS is a fault tolerant distributed system that can reallocate services away from failed member hardware units (or failed member ICs)' [column 13 line 63 to column 14 line 10]. Therefore, as shown in figure 4B, the server is doing load balancing for clients 495.

F. *The Appellants argue that in direct contrast to the elements of claim 21, Abjanic makes clear that the described apparatus is an application-layer content-based switching apparatus.*

The examiner respectfully disagrees. Abjanic teaches in figure 4 is a block diagram illustrating a traffic manager according to another example embodiment. Traffic manager 140 includes a security accelerator 415 for encrypting outgoing messages and/or decrypting incoming messages received from the network. According to an embodiment, the security accelerator 415 is a Secure Sockets Layer (SSL) accelerator, available from Intel Corporation. The security accelerator 415 allows the security related tasks such as encryption and/or decryption to be off-loaded from the application server to the accelerator 415 of the traffic manager 140 [column 10, lines 1-12]. So as illustrated in figure 4, there is a security accelerator that is a SSL accelerator.

The examiner points out to the board that the method for performing acceleration of data communications between a plurality of customer devices in the present application (page 23 of the present application) is a SSL acceleration device.

G. *The Appellants argue that Abjanic fails to teach or suggest mechanisms by which a load balancing acceleration device can, without processing the data packets with an application layer of a network stack, selecting with the acceleration device at least one of the plurality of servers in the enterprise based on a load calculation including processing sessions of other servers in the enterprise and associating the selected server with a communications session from one of the clients.*

The examiner respectfully disagrees. Abjanic teaches that traffic manager 140 also includes a director 145B and a broker 410. A decrypted message is received by broker 410 from security accelerator 415. According to an example embodiment, broker 410 operates as both an output interface (similar to output interface 320) and a load balancer to balance or adjust the traffic among one or more of servers or processing nodes within the data center 135 [column 10, lines 13-19].

This section teaches that the traffic manager has capabilities of acceleration and load balancing. The Abjanic reference was not used to teach bypassing the application layer. The Toporek reference was used to teach that feature. Abjanic teaches the data center 135 is provided for sending, receiving and processing a wide variety of messages, requests, business transactions, purchase orders, stock quotes or stock trades, and other information. The data center 135 includes several processing nodes (e.g., servers), including server 150, server 160 and server 170 for handling the various orders, business transactions and other requests. The different servers in data center 135 may be allocated to provide different services, or even different levels of services. According to an example embodiment, the clients and the data center 135 exchange business transaction information or other information by sending and

receiving XML messages (data provided in XML or in a XML based language), or messages based upon another type of structured syntax for data interchange. The various servers (e.g., servers 150, 160 and 170) are coupled to a traffic manager 140 via a switch 165. Traffic manager 140 may perform a variety of functions relating to the management of traffic, including load balancing (e.g., balancing the load of incoming messages or requests across the available servers according to some policy, such as round-robin, least number of connections, or other load balancing technique). Referring to the clients again in FIG. 1, application program 112 may be a business program or a program for managing inventory, orders or other business transactions. For example, application program 112 may automatically and electronically detect that inventory has decreased below a threshold value and then automatically generate and send a purchase order to a supplier's server at data center 135 to request a shipment of additional supplies or inventory. Thus, server 110 may initiate, for example, a business-to-business (B2B) transaction by sending an electronic order to the supplier's remote server located at data center 135 [column 3 line 66 to column 4 line 32]. Therefore, as taught by Abjanic, various severs will do load balancing for incoming messages to other servers that are in communication with clients.

H. *The Appellants argues that the combination of Abjanic in view of Toporek does not suggest that secure communications would be processed any differently whatsoever.*

The examiner agrees. However, it is not claimed in claim 12 that that secure communications would be processed any differently whatsoever.

I. *The Appellants argues that the Baskey approach explicitly requires that secure data travel two full networking stacks, including the application layer, and Baskey fails to describe any other mechanism for implementing the SSL proxy.*

The examiner respectfully disagrees. As discloses in the abstract of the Baskey reference, two different connections are formed. Baskey teaches a first session specific SSL connection, different from the persistent secure connection, is also established between a first client application and the SSL proxy server. Communications between the first client application and the SSL proxy server transmitted over the first session specific SSL connection are then

forwarded with the client's identity preserved to the transaction server over the persistent secure connection. Furthermore, a second session specific SSL connection between a second client application and the SSL proxy server may also be established and the communications between the second client application and the SSL proxy server transmitted over the second session specific SSL connection are forwarded to the transaction server over the persistent secure connection. Preferably, the persistent secure connection is an SSL connection. Both connections are SSL connections. The present application employs SSL connections as well.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Aravind K Moorthy

Conferees:

Taghi Arani

Eddie Lee

EDDIE C. LEE
SUPERVISORY PATENT EXAMINER